Day 3 overview

- Making scripts
- Understanding compute clusters
- Using Slurm with SBATCH scripts
- Troubleshooting SBATCH scripts
- Transferring and downloading files

Make a script

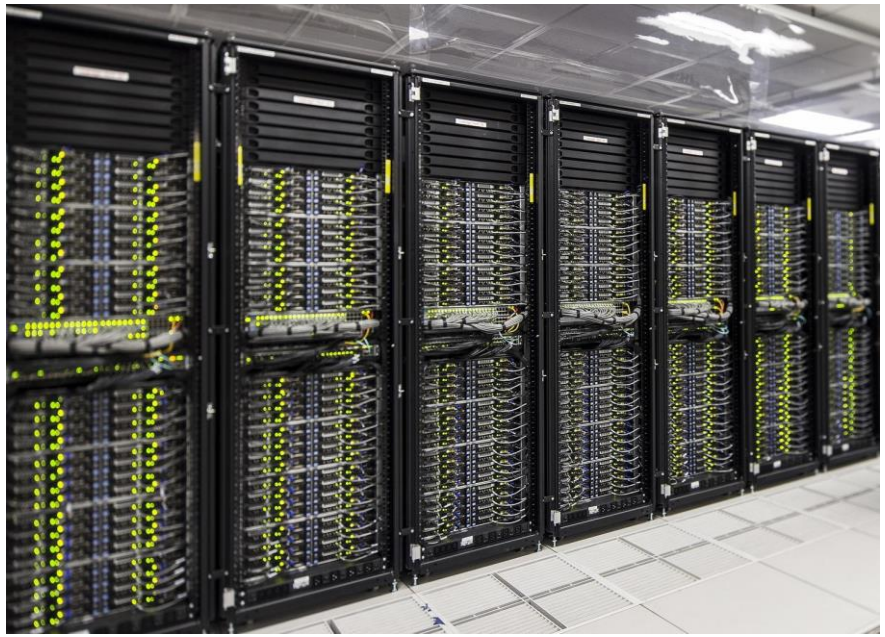
- In a terminal on your local machine:
  - Open a new file called `test_script.sh`
  - Add the following four lines:

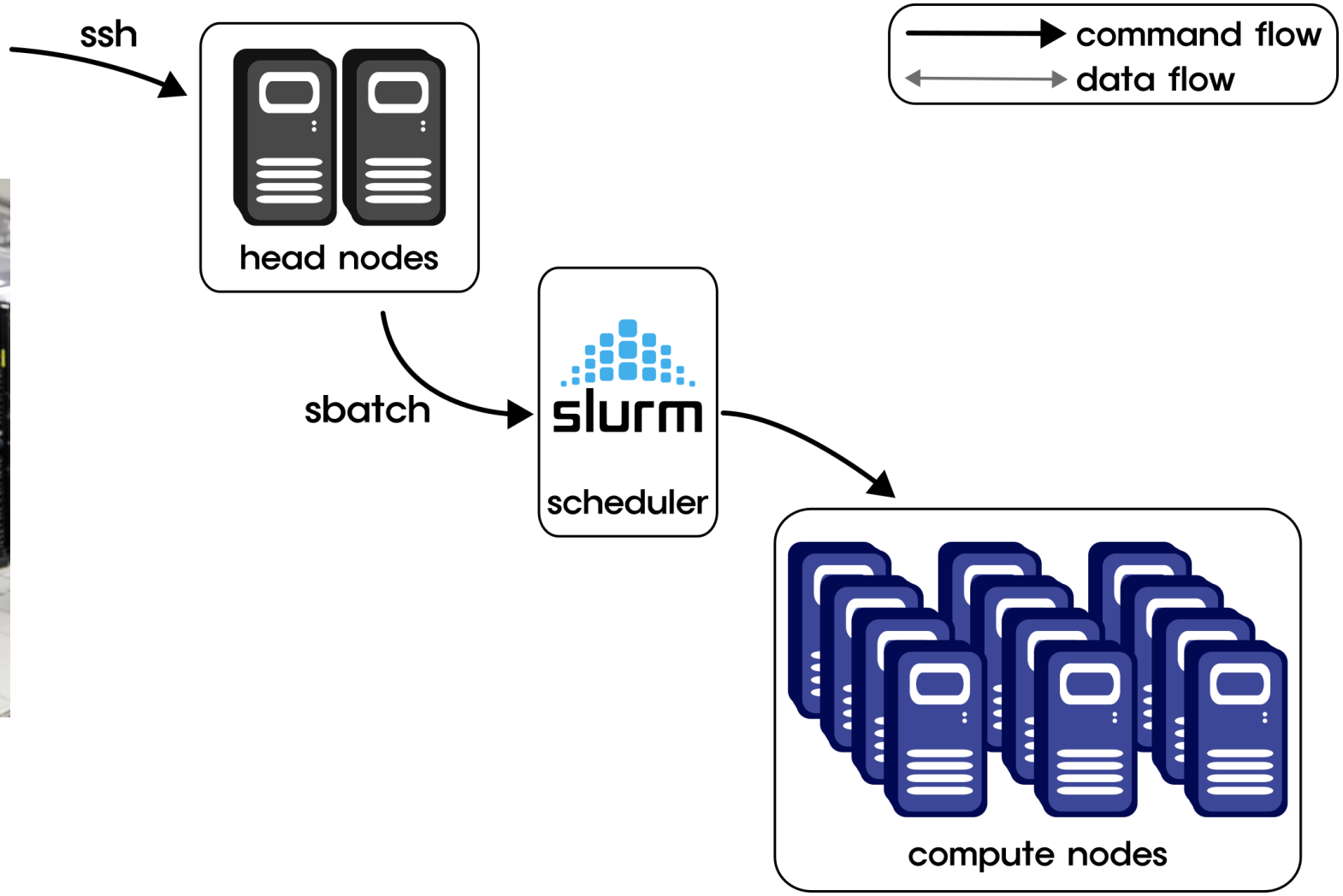
```
#!/bin/bash
pwd
sleep 3
echo “This is a test script”
```

  - Save and exit, then run the script with:

```
sh test_script.sh
```

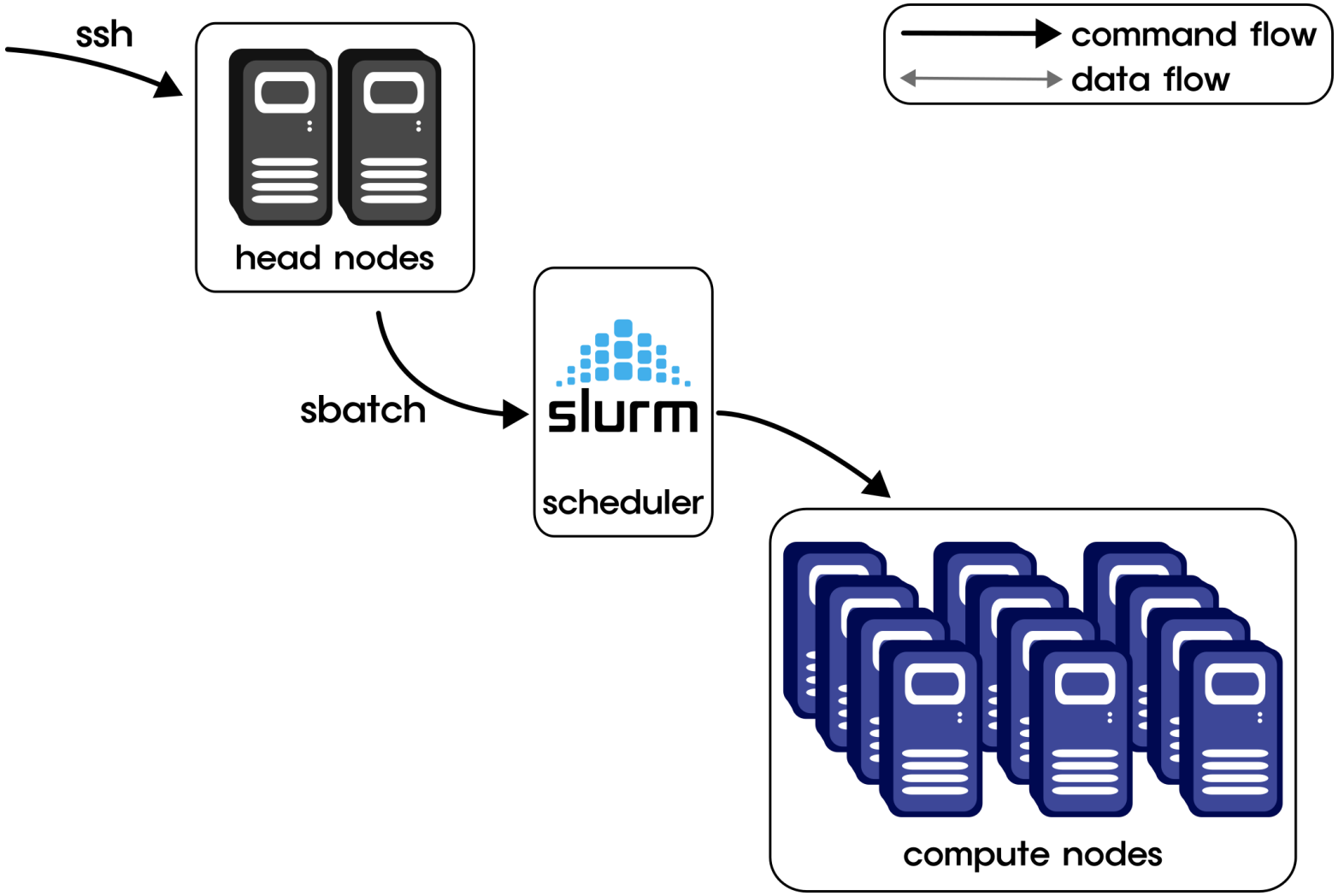
Compute cluster architecture





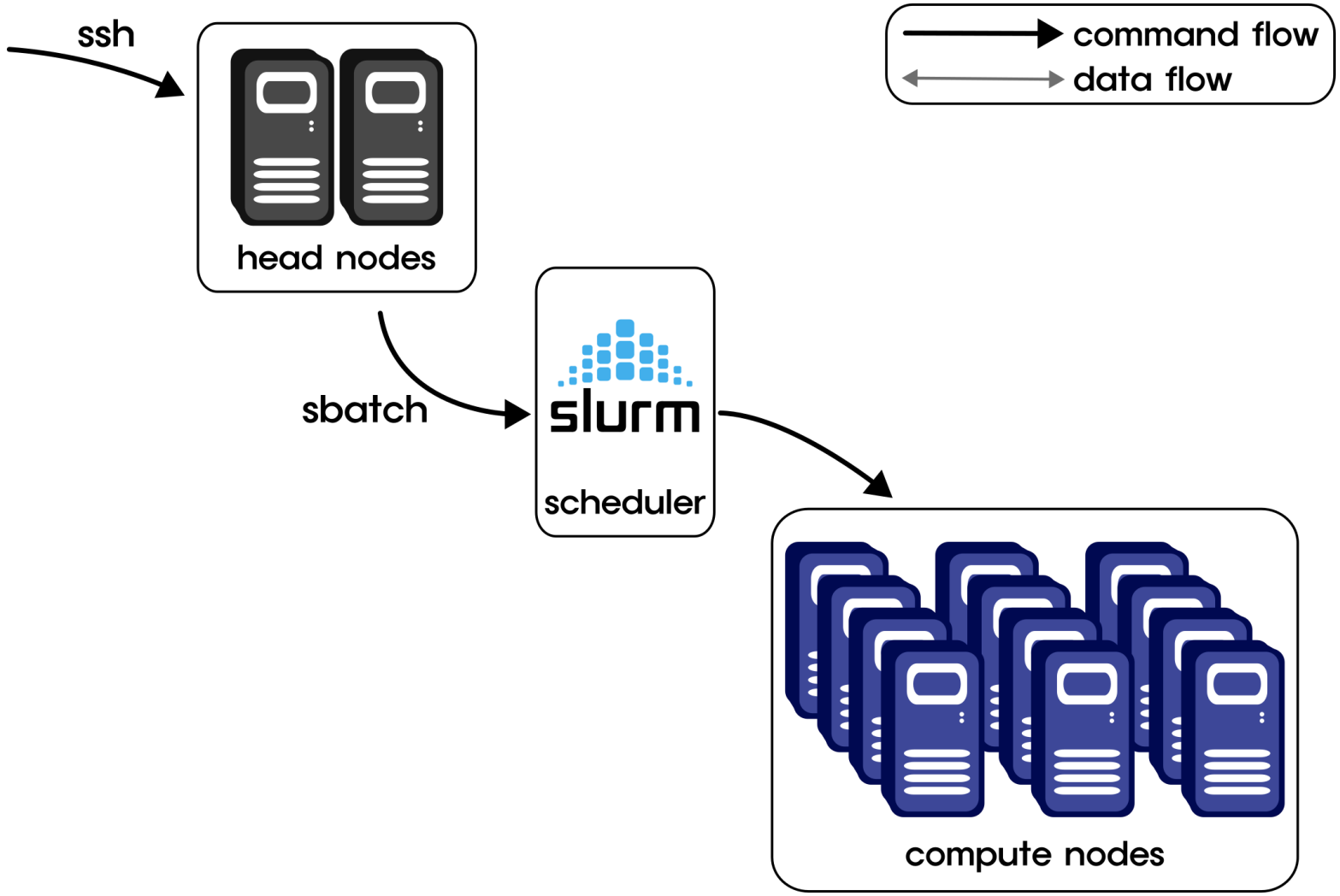
Compute cluster architecture

- Head nodes
  - Very limited resources
  - Only intended for basic interfacing/commands
- Compute nodes
  - Extensive resources
  - Take job scripts (lists of commands)
  - Require a scheduler to manage resources



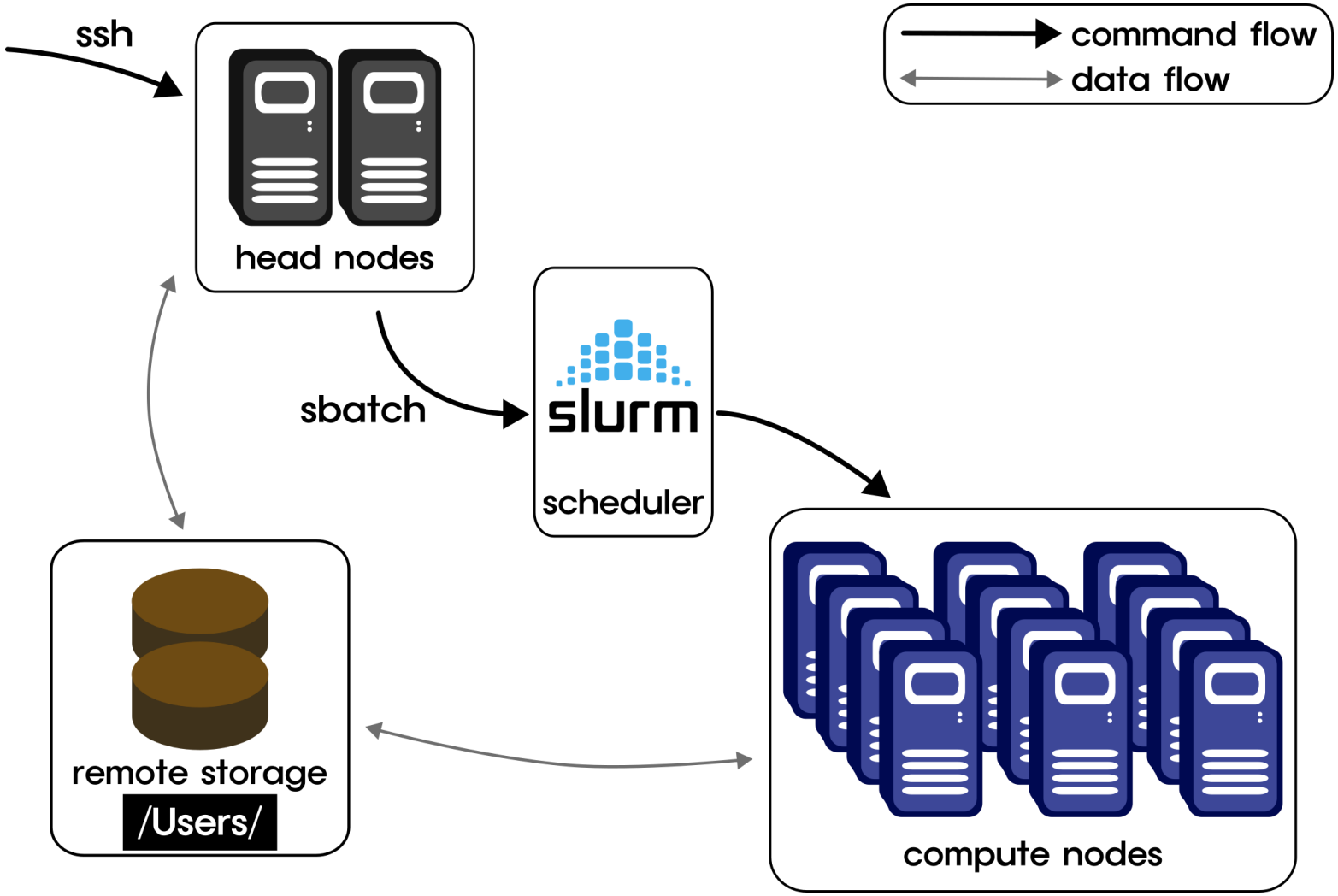
Compute cluster architecture

- Slurm (scheduler)
  - Manages resources to execute jobs
  - Executes jobs as resources become available



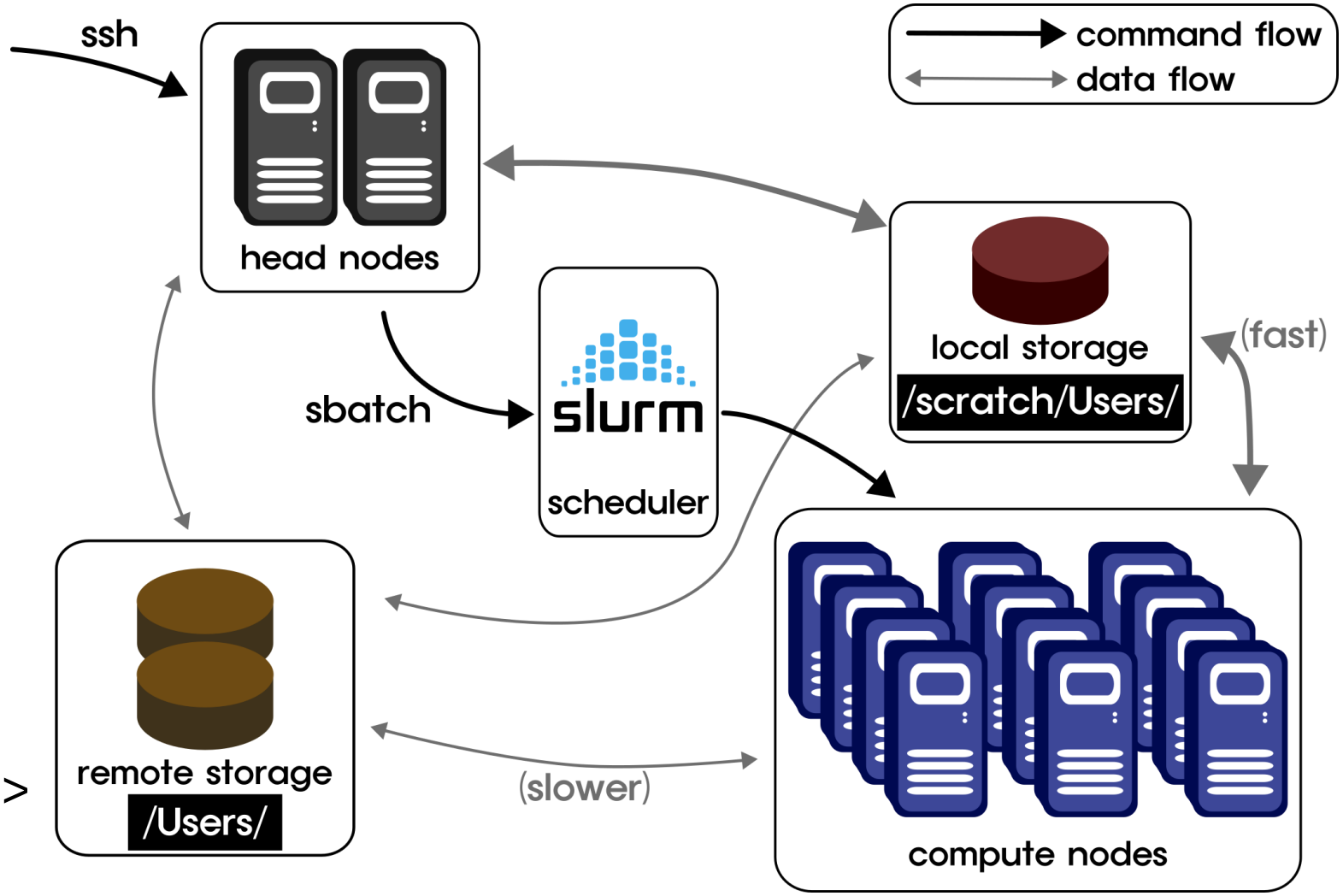
Compute cluster architecture

- Remote storage
  - Off-site from cluster (slow)
  - Backed up
  - For Fiji, located under /Users/<username>



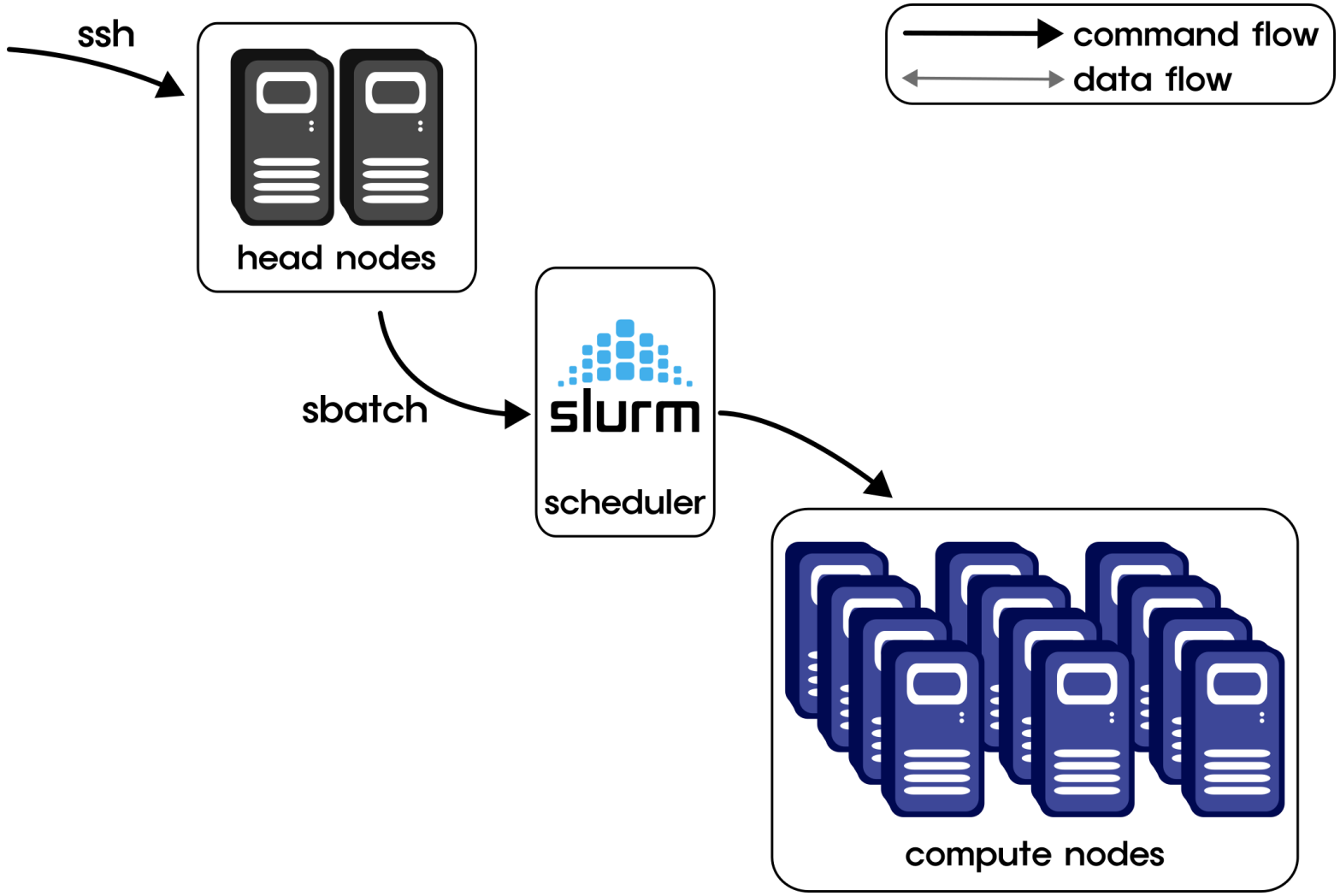
Compute cluster architecture

- Remote storage
  - Off-site from cluster (slow)
  - Backed up
  - For Fiji, located under /Users/<username>
- Local storage
  - On-site with cluster (fast)
  - NOT backed up, $$$
  - For Fiji, located under /scratch/Users/<username>



Run a job!

- Day 3 worksheet Part 1
  - Create an SBATCH script with a header and a command
  - Run the script as a job
  - Troubleshoot



SBATCH header

```
#!/bin/bash
#SBATCH -p <partition>
#SBATCH --time=00:00:00
#SBATCH --mem=<memory>
#SBATCH --ntasks=
#SBATCH --job-name=<jobname>
#SBATCH --mail-type=ALL
#SBATCH --mail-user=<you@email.com>
#SBATCH --error=<path>/%x_%j.err
#SBATCH --output=<path>/%x_%j.out
```

→ specify job partition (queue) to run the job

→ hrs:minutes:seconds

→ how many cores/processors are needed for the job

→ give the job a name

→ specify name and location of error file

→ specify name and location of log file

Output/error files are primarily used for troubleshooting:

- These files capture STDERR and STDOUT that isn't captured within the script
- %x and %j are Slurm variables that will be **automatically** filled into the filenames
- %x is the job name. If not specified in the header, it will be the name of your sbatch script file
- %j is the unique job ID assigned to your job by Slurm
- Using these variables ensures that you'll always have unique filenames for every job

- How to know when a job fails
  - Expected files are empty or missing
  - Job finishes much quicker than expected
  - Errors appear in your error file
- Why does a job fail?
  - Incorrect command/parameters
  - Incorrect paths
  - Incorrect SBATCH header
  - Incorrect file format
  - Issues with software versions

BREAK

Understanding Modules

- Environmental variables alter how we interact with the cluster or help us “find” commands
  - An important environmental variable is `$PATH`, which specifies where the computer looks for commands

    (look at it with `echo $PATH`)
- Modules are set up by admin and allow you to easily change environmental variables
  - Not present on every compute cluster
  - Modules are automatically unloaded after a session terminates

Module commands

`module avail`	List available modules
`module spider`	Describe modules (with tab complete)
`module load <module>`	Load specific module
`module list`	List currently loaded modules
`module unload <module>`	Unload specific module
`module purge`	Unload all current modules

Analyze data!

- Day 3 worksheet Part 2
  - In your SBATCH script, write a command to run the program FastQC
  - Run the script as a job
  - Troubleshoot if necessary
  - Transfer the HTML output to your local computer and view it

Homework for day 4

- Videos for day 3 (if not already done)
- Videos for day 4
- Homework_day3.md worksheet